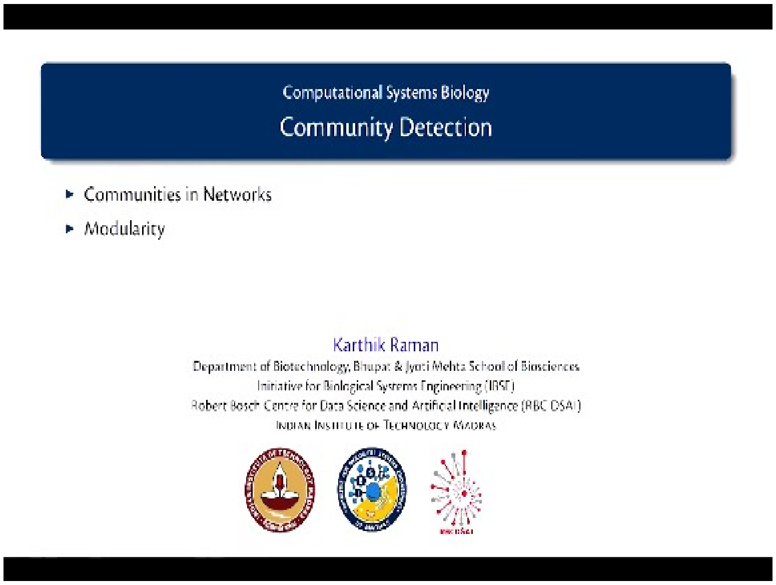
Computational Systems Biology

Karthik Raman
Department of Biotechnology
Indian Institute of Technology - Madras

Lecture – 23
Community Detection

(Refer Slide Time: 00:11)



Computational Systems Biology
Community Detection

- Communities in Networks
- Modularity

Karthik Raman
Department of Biotechnology, Bhupat & Jyoti Mehta School of Biosciences
Initiative for Biological Systems Engineering (IBSE)
Robert Bosch Centre for Data Science and Artificial Intelligence (RBC DSAI)
Indian Institute of Technology Madras

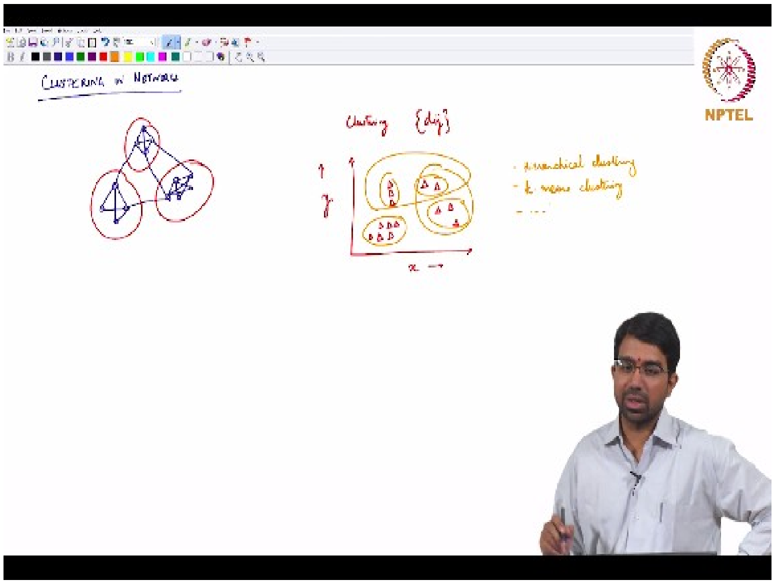
In today's video, we will have a brief glimpse of community detection which is a very important topic in graph theory and, you know, has attracted work from a lot of computer scientists and mathematicians in the past. We will look at communities in networks, more from biological perspective and the concept of modularity.

(Refer Slide Time: 00:29)

COMMUNITIES IN NETWORKS

So next, let’s understand communities in networks.

(Refer Slide Time: 00:34)



Clustering in Network

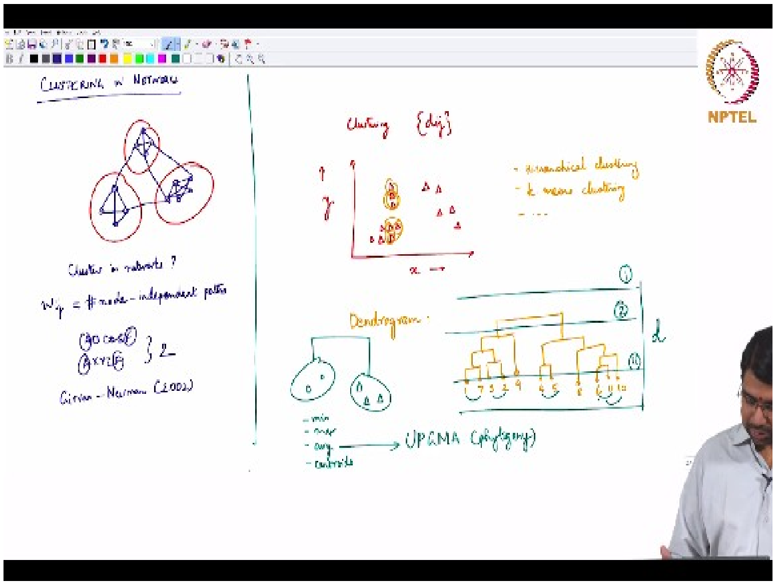
So the other name for studying this is known as clustering. What do we understand by a cluster? So higher intraconnectivity, lesser interconnectivity. So, if I had, I might say that there are 3 clusters in this graph. So, we will do a quick detour to see, to understand what clustering is in the first place. Clustering essentially outside of graph theory. What do we mean by clustering, right? So there are different ways to cluster.

So this is a classic method in machine learning, in fact in unsupervised learning wherein we want to discern some patterns in any given dataset. So, you are given a particular dataset, you want to group similar items together, you want to cluster similar items together. Usually most clustering methods work using a distance matrix, right. You have a matrix of distances and based on that matrix you try to identify which nodes are closer to one another and so on.

So, let's take a very simple example. So, let's say you have some x and some y and you have different data points. So you may end up clustering this in different ways. So, 1 possible clustering could be or may be this becomes a cluster or may be this becomes a cluster, so you could have different types of clustering. So, there are few popular algorithms for clustering.

So, there is hierarchical clustering and k-means clustering and several other methods. Let's not get into that at this moment but let's look at the example of hierarchical clustering because it is very germane to our current problem. What is hierarchical clustering? You essentially first find out the 2 nodes that are closest.

(Refer Slide Time: 03:46)



So, given a set of points like this, let's get rid of the cluster, given these points, which 2 points are closest to one another, right. To me, it looks like these 2 points might be very close to each other followed by these 2, these 2, these 2, then may be these 2, these 2 and so on, right. So we can get a ranking of which ij's are closest to one another and you then start building what is known as a dendrogram.

Something of this sort. So the leaves of the dendrogram are all your data points, right. So, let's say 1 7 3 2 9 4 5 8 6 11 10 something like that whatever are your points. I haven't exactly mapped it to this figure but what this gives you is, you can cut at different levels to get different clusterings. Let's think of this as some distance axis, right. I cut here, I get 1 cluster. I cut here, I get 2 clusters. I cut here, I get 11 clusters, right. Every node is a, every data point is a cluster by itself, okay. So this is how hierarchical clustering works.

The dendrogram is built by using distances, right. You first join the 2 nodes that are closest to each other. So, in this example, it looks like 1 and 7 were very close to each other as were 6 and 11 as were 4 and 5 or 2 and 3. The next closest was perhaps this cluster and 10, right. So, let's just look at this example. How do you compute the distance between these 2 clusters? You want to know whether to hook these up in a dendrogram of this sort, right. How would you compute the distance between these 2 clusters?

Distance between 2 points, no problem, you can compute the distance. You will compute some, some Eucledian distance or any Hamming distance or many different kinds of distances are there, you could pick one of them that make sense. How would you now compute the distance between 2 sets? So, you can have multiple, you can have minimum distance, you can have maximum distance, you can have average distance, or you could take centroids, right.

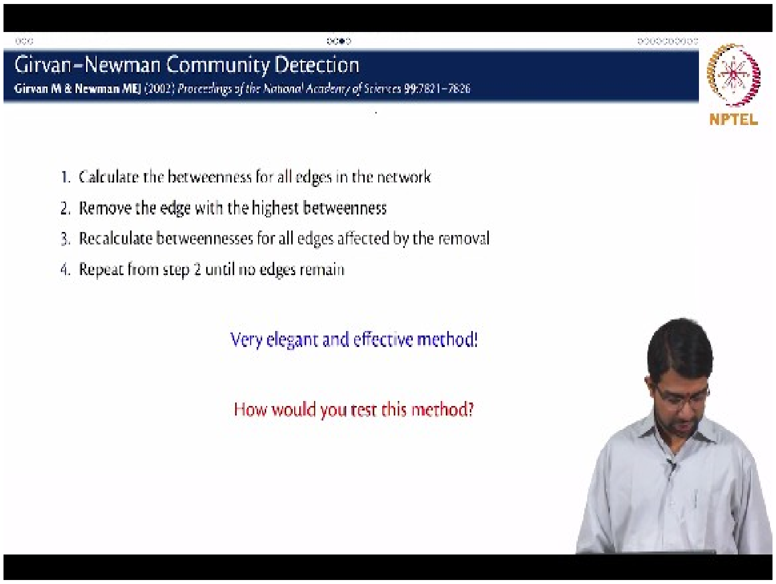
Each of this has a different name. So, you have single link algorithm, complete link algorithm, central linkage algorithm and so on. So each of these have a different name. So for those of you who studied bioinformatics before and so on, you may be familiar with this term. This basically means unweighted pair group mean average which is nothing but this, right. This is used in phylogeny or phylogenetic reconstruction.

Anyways, so this was the aside. So what is the main topic that we are looking at, how do we do clustering in networks? Can you think of an idea in the light of what we see on the right-hand side? In fact, the classic way to do this is, you compute another matrix called W, right, where Wij measures the number of, number of node-independent paths. What is a node-independent path?

So if you have ABCDEF and AXYZF, these are node independent paths. There is no similarity between these 2 paths except the terminus, right. So now you will say that these 2, this pair of nodes are connected by 2 paths, is connected by 2 paths. You use these as weights to create a matrix and then you apply the regular clustering business.

Another very popular algorithm for clustering in networks was given by Girvan and Newman. So you will see Newman's name appearing a lot of times in this part of, part of the course because Newman is one of the top network scientists. So they had an algorithm which actually uses edge betweenness. So, let's see how that algorithm works in a moment.

(Refer Slide Time: 10:25)



Girvan-Newman Community Detection

Girvan M & Newman MEJ (2002) *Proceedings of the National Academy of Sciences* 99:7821–7826

1. Calculate the betweenness for all edges in the network
2. Remove the edge with the highest betweenness
3. Recalculate betweennesses for all edges affected by the removal
4. Repeat from step 2 until no edges remain

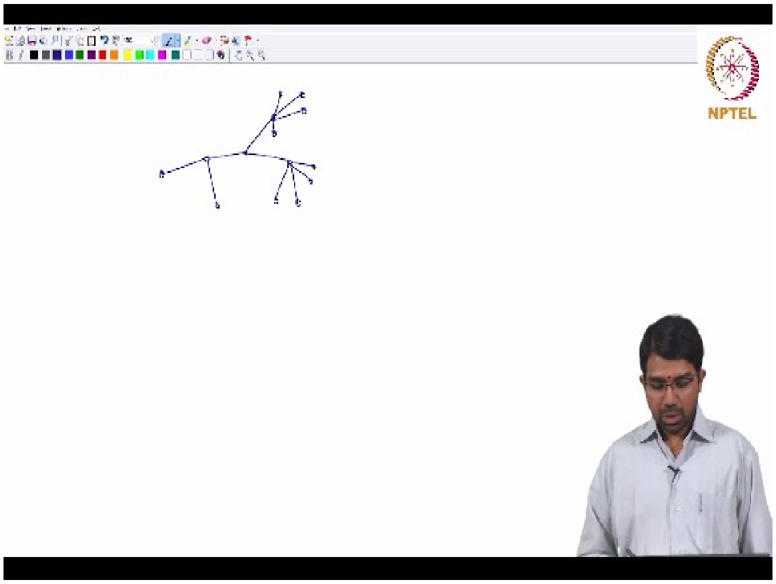
Very elegant and effective method!

How would you test this method?

So how does the Girvan-Newman algorithm works? You first calculate betweenness for all edges in the network. You then remove the edge with highest betweenness. Recalculate betweenness for all the edges that have been affected by the removal, repeat. What will you have at the end? All nodes and no edges, right. But if you, if you rewind the process, if you play the whole process backwards, you will find that nodes get connected pair by pair.

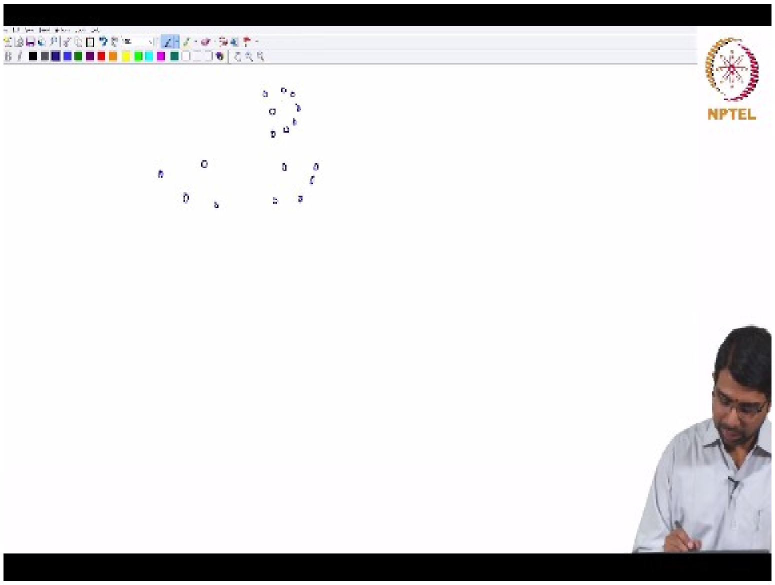
You will first add one edge, then you will add another edge. This is equivalent of hierarchical clustering in some sense. Can you imagine this? Let's see if we can visualize it.

(Refer Slide Time: 11:24)



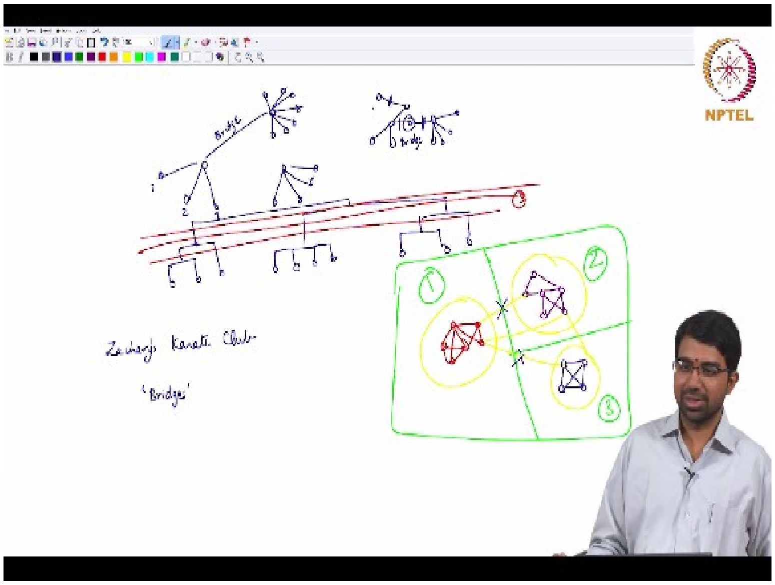
This is too symmetric. So, which edge will have the highest betweenness here?

(Refer Slide Time: 11:48)



This one, perhaps, right. So remove that first, right. Following this, it may be this one, you remove it. Following this, basically everything has very similar edge betweennesses, so remove everything. There was a node here, right and then you remove off all of these edges.

(Refer Slide Time: 12:35)



So now let's start putting back the edges. We first add these edges, then we add these edges, then we add these edges and this would correspond to in the hierarchical clustering. So, let's say this is 1 2 3, something of this sort and then these get added or in fact, you can have it very well like this. They are all the same distance and may be something else like this here. Then this gets connected to this, this gets connected to this.

Now you have a hierarchical clustering. You know where to split the communities. If you cut, if you cut here, you get one single community. If you cut here, you get 4 or 5 communities. If you cut here, you will get 3 communities. You have to judge what seems more like a community. How would you test this sort of an algorithm? I give you multiple algorithms for community detection.

I said you compute node-independent paths and then try to identify what are the communities or you use the Girvan-Newman community algorithm. How will you judge whether the algorithm is performing well? What would be the test case to study the algorithm and its performance? So, you would start with a sort of community and make some more communities. All these look like communities, no?

And now add edges to these communities. Add edges such that you add more intracommunity edges and not any inter, no fewer, you don't even worry or you already have enough intracommunity edges. You just add, throw in a few intercommunity edges. Now, if you throw this community back at your algorithm, it should fish out these communities again, right. So there are 3 communities here. So, in fact this is how Girvan and Newman tested their algorithm and reported it.

They also demonstrated it on certain popular networks. One of them is this interesting story about Zachary's karate club. What happened was this was a popular karate club where there was some infighting and finally this club split into 2 but they could actually use the social network of the club and actually predict how the split would have happened, right. They showed that this, this split wasn’t surprising once the split happened.

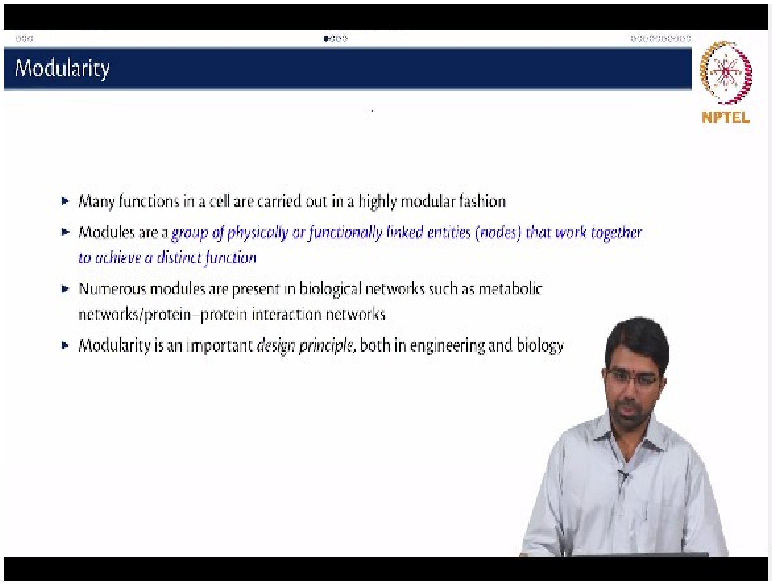
Because they knew the social connectivity of the members and then they computed this, they did this community detection and for them this was the most likely split that was going to happen. It was indeed what had happened. Yes, I think British were very good at this right, so they divide and conquer. You know what are all the weak links that join the communities and you go and first disable these links, then you can easily separate out the communities. So what is the first stage to remove? What we just saw?

The one with highest betweenness. As long as they can find, you can easily disintegrate the network. In fact, there is also this related concept of bridges. You have the concept of a bridge node or a bridge edge. Bridge edge is one, that edge which is a bridge between 2 groups, right. So, you remove the bridge edge, you end up with 2 connected components. So, this would be a bridge edge and in this case, this would be a bridge node. **“Professor - student conversation”** There is no bridge node. You may have both, right. So, here this would be a bridge edge, right and this would be the bridge node. So, what I have circled is the bridge node but this is potentially a bridge edge or this, even these are all bridges, right but they are not very important bridges.

They don’t, so you can look at what bridge gives you the biggest cut, right, the most equal cut. These are in fact very important problems that are studied for example in, in fact in our own chemical engineering wherein, how do I find leaks in a water network and so on, right? Or where do I need to make the measurements to predict leaks?

So, you have a big network of water distribution and where do I place sensors appropriately so that I can make the best measurements to find out where the leaks are and so on and so forth. There is in fact a graph theory course out of chemical engineering, right. That is the reason why chemical engineers are very interested in graph theory, okay. This is a very elegant and effective method and I have already asked you how you would test this method.

(Refer Slide Time: 18:54)



Modularity

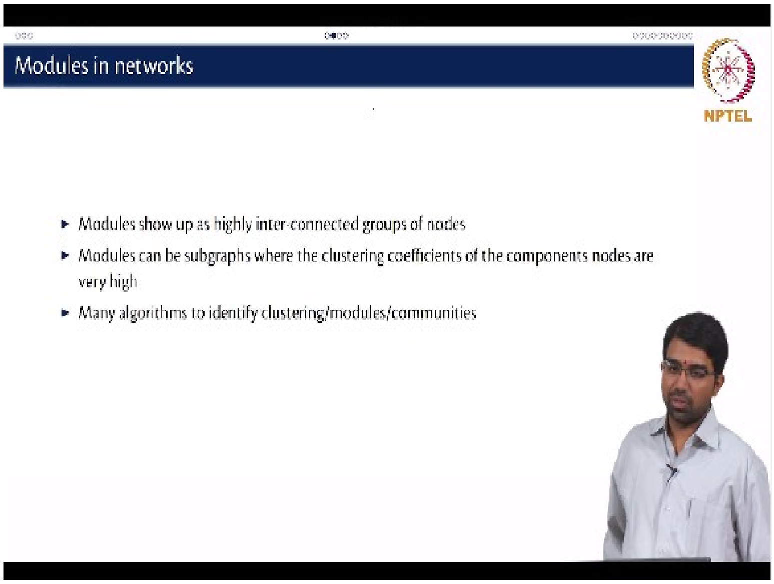
- Many functions in a cell are carried out in a highly modular fashion
- Modules are a *group of physically or functionally linked entities (nodes) that work together to achieve a distinct function*
- Numerous modules are present in biological networks such as metabolic networks/protein–protein interaction networks
- Modularity is an important *design principle*, both in engineering and biology

So I then introduced modularity to you. I directly went into how we calculate modularity and so on. Why is modularity important? It is because many functions in a cell are carried out in a highly modular fashion and what is a module? It is basically a group of physically or functionally linked nodes that work together to achieve a particular function. And there are many modules in different types of networks, be it a gene regulatory network or a signalling network or a protein interaction network and so on.

You could think of pathways in metabolic networks being modules. It's an important design principal both in engineering and biology. It helps in some sort of decoupling and it helps insulate different functions. Many kinds of useful outcomes are there for modularity and there is a very interesting study that was performed by Kashtan and Alon in, I think, 2003 or may be the late 90s where they showed that if you have a gene regulatory network and you subject it to alternating stresses, the network evolves to become more modular because then it can do nice switching, right.

So, if you have a really, you know, complex network connectivity and there are multiple stresses, the so much firing that has to happen to respond to that stress. But if you have nice modules, then in one stress this module can become active and in the other stress, the other module can become active. So, they found that networks started reorganizing themselves into nicer modules when alternating stresses were provided and so on.

(Refer Slide Time: 20:30)

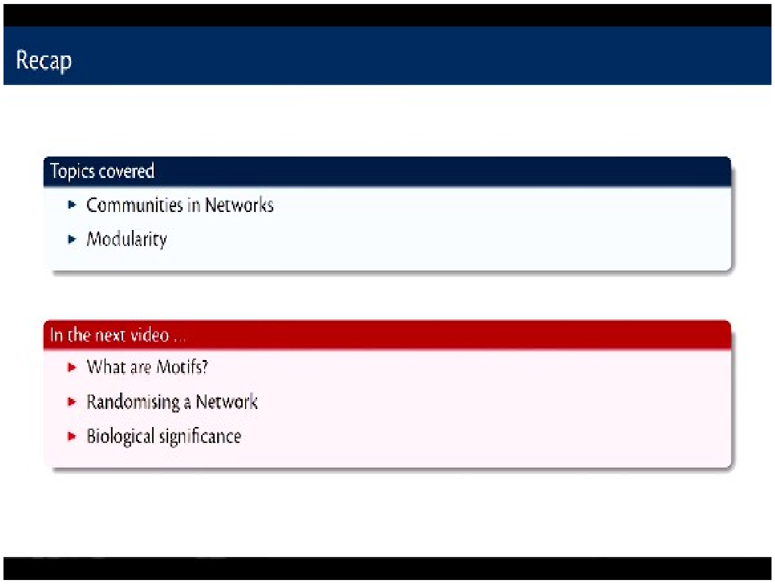


Modules in networks

- Modules show up as highly inter-connected groups of nodes
- Modules can be subgraphs where the clustering coefficients of the components nodes are very high
- Many algorithms to identify clustering/modules/communities

So, modules essentially show up as highly interconnected groups of nodes. You could think of them as subgraphs where the clustering coefficients of the nodes are very high. In other words, they are cliques or near cliques. So, 0.8 is already clique enough. And there are many algorithms to identify clustering in modules or clustering of modules or communities and we did look at couple of these methods.

(Refer Slide Time: 20:55)



Recap

Topics covered

- Communities in Networks
- Modularity

In the next video ...

- What are Motifs?
- Randomising a Network
- Biological significance

So, in today's lecture, we looked at communities in networks and the concept of modularity. In the next video, we will study what are motifs? How do you identify motifs for which you need to identify a null model which is usually obtained by randomization and so on and what is the biological significance of the existence of motifs, right? What does it mean to have motifs in a biological network.